

Bases de Datos Espacio-Temporales aplicadas en la gestión de emergencias

Marcela Aballay

Maestría en Ing. en Sist. de Información
FRC - UTN
Córdoba, Argentina
marcelaaballay@yahoo.com.ar

Anabella De Battista

Depto. en Ing. en Sist. de Información
FRCU - UTN
Entre Ríos, Argentina
debattistaa@frcu.utn.edu.ar

Edilma O. Gagliardi

Departamento de Informática
Univ. Nac. de San Luis
San Luis, Argentina
oli@unsl.edu.ar

Resumen—En este trabajo se presenta la implementación de un índice métrico-temporal en una aplicación que tiene como objetivo gestionar móviles destinados a la atención de emergencias en la vía pública. Esta aplicación fue desarrollada en el marco de una tesis de maestría para el Municipio de la localidad de Córdoba, Argentina. Se detalla la implementación del índice, la combinación del modelo de bases de datos relacional y espacio-temporal y la resolución de las principales consultas requeridas para la aplicación, fundamentalmente la consulta de trayectoria de los móviles en la atención de un evento.

Palabras Claves—I+3 R-Tree, bases de datos espacio-temporales, gestión, emergencias, móviles

I. INTRODUCCIÓN

El modelo relacional de bases de datos permite gestionar información estructurada, es decir, información que puede organizarse en registros, cuyos campos son totalmente comparables. En los últimos años han surgido aplicaciones que requieren gestionar información que no se puede estructurar, por lo que es necesario contar con nuevos modelos de bases de datos que cuenten con métodos que permitan realizar su gestión de manera eficiente. En algunos casos resulta de interés manejar objetos que tienen asociados atributos como el espacio y el tiempo. Estos objetos conforman el tipo de dato espacio-temporal y, como soporte para el tratamiento de este tipo de datos, surgen los Métodos de Acceso Espacio-Temporales (MAETs), cuyo principal objetivo es evitar que se deba realizar un examen exhaustivo de la base de datos al momento de responder consultas, además de brindar eficiencia ante consultas que se realizan con alta frecuencia. En este artículo se presenta el desarrollo de una aplicación que implementa el índice espacio-temporal *I+3 R-Tree* para el manejo de objetos en movimiento. La aplicación tiene como objetivo administrar móviles para la atención de emergencias en la vía pública. El trabajo se organiza de la siguiente manera: en la Sección 2 se presenta el marco teórico, en la Sección 3 se presenta la aplicación desarrollada y en la Sección 4 las Conclusiones.

II. MARCO TEÓRICO

A. Bases de Datos Espaciales

Este modelo de Bases de Datos permite procesar objetos con alguna referencia espacial y que poseen normalmente una estructura compleja. Un dato espacial puede ser un punto, una poligonal o un polígono [1], [2]. Con un punto podrían

representarse automóviles, edificios, estaciones de trenes, etc.; las poligonales se utilizan para modelar ríos, rutas, cableados, etc., y los polígonos se pueden utilizar para representar países o regiones. El principal ámbito de aplicación de las bases de datos espaciales es el de los Sistemas de Información Geográfica (SIG) [3]. La recuperación y actualización de los tipos de datos espaciales se basan no sólo en el valor de ciertos atributos, sino también en la ubicación espacial del objeto. Por ejemplo, nos podría interesar obtener los terrenos geográficamente adyacentes a uno dado, o encontrar todas las estaciones de servicio por las que pasa una ruta. Este tipo de bases de datos proporciona la estructura capaz de soportar datos espaciales, además de proveer un lenguaje de consulta y métodos de acceso eficientes [4].

B. Consultas Espaciales

El atributo espacial (o geometría) de los objetos se encuentra definido en un espacio euclideo n -dimensional. Siendo o un objeto espacial, $o.G$ un atributo espacial y $o.G \subseteq \mathbb{R}^n$ un predicado espacial, a continuación se describen las principales consultas que pueden realizarse a una Base de Datos Espacial [3], [5], [6]:

- EMQ (Exact match query): dado un objeto o' encuentra todos los objetos o iguales en forma, posición y tamaño.

$$EMQ(o') = \{o \mid o'.G = o.G\}$$

- PQ (Point query): encuentra los objetos o que se superponen o cubren al punto consultado p .

$$PQ(p) = \{o \mid p \cap o.G = p\}$$

- WQ o RQ (Window query o range query): dado un rango $q \in \mathbb{R}^n$, encontrar todos los objetos o que tienen al menos un punto en común con q .

$$WQ(q) = \{o \mid q \cap o.G \neq \emptyset\}$$

- IQ (Intersection query, region query u overlap query): dado un objeto o' con un atributo espacial $o'.G \in \mathbb{R}^n$ devuelve los objetos o de la base de datos que tienen al menos un punto en común con o' .

$$IQ(o') = \{o \mid o'.G \cap o.G \neq \emptyset\}$$

- EC (Enclosure query): dado un objeto o' con un atributo espacial $o'.G \in \mathbb{R}^n$, encontrar todos los objetos o encerrados por o' .

$$EQ(o') = \{o \mid o'.G \cap o.G = o'.G\}$$

- CQ (Containment query): dado un objeto o' con un atributo espacial $o'.G \in \mathbb{R}^n$, devolver los objetos o de la base de datos que están encerrados por o' .

$$CQ(o') = \{o \mid o'.G \cap o.G = o.G\}$$

- AQ (Adjacency query): dado un objeto o' con un atributo espacial $o'.G \in \mathbb{R}^n$, devolver todos los objetos o adyacentes a o' .

$$AQ(o') = \{o \mid o.G \cap o'.G \neq \emptyset \wedge o'.G^0 \cap o.G^0 = \emptyset\}$$

En este caso $o.G^0$ y $o'.G^0$ denotan el interior de las figuras geométricas formadas por los atributos espaciales $o.G$ y $o'.G$ respectivamente.

- NNQ (Nearest-Neighbor Query): devuelve los objetos de la base de datos que tienen la distancia mínima al objeto de consulta. En este caso la distancia se define como la distancia entre sus puntos más cercanos (Euclidiana o de Manhattan).

$$NNQ(o') = \{o \mid \forall o'' : dist(o'.G, o.G) \leq dist(o'.G, o''.G)\}$$

- SJ (Spatial join): dadas dos colecciones de objetos espaciales R y S y un predicado binario espacial θ , encontrar todos los pares de objetos $(o, o') \in R \times S$ donde $\theta(o.G, o'.G)$ es verdadero.

$$R \bowtie_{\theta} S = \{(o, o') \mid o \in R \wedge o' \in S \wedge \theta(o.G, o'.G)\}$$

C. Bases de Datos Temporales

Este modelo de base de datos incorpora al tiempo como una dimensión, por lo que permiten asociar tiempos a los datos almacenados [7], [8]. De acuerdo a la forma en que manejan el tiempo se puede establecer una clasificación de las bases de datos temporales [9]:

- *de tiempo transaccional*: donde el tiempo se registra de acuerdo al orden en que se procesan las transacciones.
- *de tiempo vigente*: que almacenan el tiempo en que el hecho ocurrió en la realidad, que puede no coincidir con el momento de su registro.
- *bitemporales*: que integran la dimensión transaccional y la dimensión vigente a través del versionado de los estados, es decir, cada estado se puede modificar para actualizar el conocimiento de la realidad pasada, presente o futura, pero esas modificaciones se realizan generando nuevas versiones de los mismos estados.

Es decir que el tiempo válido identifica el momento de tiempo durante el cual la información es cierta en el mundo modelado y el tiempo transaccional indica el momento que los datos fueron incorporados a la base de datos [1]. Si

se realiza una consulta teniendo en cuenta el tiempo vigente esta sería "encontrar todos los móviles activos para un tiempo determinado"; en cambio si tomamos como parámetro el tiempo transaccional se consultaría "encontrar todos los móviles activos registrados en un tiempo determinado".

D. Modelo de Bases de Datos Espacio-Temporales

Permiten la gestión eficiente de objetos espaciales que cambian su posición o forma a lo largo del tiempo [10], [11] y poseen ciertas características distintivas [12]:

- Tienen una estructura compleja que no puede ser representada por una tupla. Un dato espacial, entre otros, puede ser un punto, una línea o un polígono.
- Tienen a ser muy grandes. Por ejemplo un mapa puede ocupar varios gigabytes de almacenamiento.
- No existe un álgebra espacial o espacio-temporal estándar, por lo tanto no hay un conjunto de operadores estandarizados.
- Se emplean operadores espaciales altamente dependientes de la aplicación en particular.
- Debido a la complejidad de los datos las operaciones espaciales implican un costo de procesamiento más alto que las operaciones relacionales.
- Por su dinamismo, requieren que la estructura de datos soporte frecuentes inserciones, eliminaciones y actualizaciones.
- Los operadores espaciales no cumplen con la propiedad de clausura. Por ejemplo, la intersección de polígonos no necesariamente da como resultado otro polígono.

Las principales consultas que incorporan predicados espaciales que involucran instantes o intervalos de tiempo son [6], [13]:

- TS(*TimeSlice*): dado un predicado espacial definido por un rectángulo $Q \in \mathbb{R}^n$ y un instante de tiempo t , reportar todos los objetos que se intersectan espacialmente con Q en el instante de tiempo t :

$$TS(Q, t) = \{o \mid o(t).G \cap Q \neq \emptyset\}$$

- TI(*TimeInterval*): dado un rectángulo $Q \in \mathbb{R}^n$ y dos valores de instantes de tiempo t_1 y t_2 , seleccionar todos los objetos que se intersectan espacialmente con Q en algún instante de tiempo entre t_1 y t_2 :

$$TI(Q, [t_1, t_2]) = \bigcup_{\{t_1 \leq t \leq t_2\}} \{o \mid o(t).G \cap Q \neq \emptyset\}$$

Si bien estos dos tipos de consultas en general han sido las de mayor interés, se han definido otros tipos de consultas:

- RST (*Reunión o join espacio-temporal*): dados dos conjuntos de objetos espacio-temporales R y S y un predicado espacial θ , se define la reunión espacio-temporal de la siguiente manera:

$$R \bowtie_{\Theta} S = \{ \langle (o, o'), (t_i, t_f) \rangle \mid t_i, t_f \in T \wedge t_i \leq t_f \wedge (\forall t \in [t_i, t_f] (o, o') \in R(t) \bowtie_{\Theta} S(t)) \}$$

- **K-NN (*k* vecinos más cercanos):** dado un objeto o' con $o'.G \in \mathbb{R}^n$, encontrar los k objetos más cercanos a o' en un instante de tiempo dado t .

$$K-NN(o', t) = \{ o \mid \forall o'' : dist(o'(t).G, o(t).G) \leq dist(o'(t).G, o''(t).G) \}$$

- **Eventos:** se obtienen todos los eventos que sucedieron en una región en un instante o período de tiempo dados.

$$Evento(Q, t, e) = || \{ o \mid e(o(t), Q, t) \} ||$$

donde $||X||$ representa la cardinalidad del conjunto X .

- **Trayectoria:** también denominada consulta sobre patrones espacio-temporales (STP). Se expresa como una secuencia θ de longitud arbitraria m de predicados espacio-temporales de la forma

$$\theta = \{ (Q_1, T_1), (Q_2, T_2), \dots, (Q_m, T_m) \}$$

donde en cada par (Q, T) , Q representa un predicado espacial y T una restricción temporal. En esta definición Q puede representar ya sea un rango espacial (window query) o una consulta por el vecino más cercano (K-NN). T puede ser un instante de tiempo (t), un intervalo de tiempo (Δt) o vacío (\emptyset). La evaluación de esta consulta permite recuperar todos los objetos que satisfacen todos los predicados espacio-temporales especificados en θ , es decir,

$$STP(\{ (Q_1, T_1), (Q_2, T_2), \dots, (Q_m, T_m) \}) = \{ o \mid o \text{ satisfice}(Q_i, T_i) \forall 1 \leq i \leq m \}$$

E. Índices Espacio-Temporales

Los objetos que se administran con bases de datos espacio-temporales poseen características dinámicas, por lo que las bases de datos espacio-temporales deben ser capaces de almacenar grandes volúmenes de información a lo largo del tiempo, y resulta imposible pensar en un procesamiento secuencial al realizar una consulta. En este contexto surgen índices que utilizan métodos de acceso espacio-temporales adecuados para la recuperación eficiente de datos.

El índice espacial R-Tree, propuesto por Guttman en 1984 [14], ha sido implementado por la mayoría de los Sistemas Gestores de Bases de Datos Espacio-Temporales (SGBDET) como Oracle, SQL y PostgreSQL. Dentro de las variantes del R-Tree se encuentran: métodos que consideran el tiempo como otra dimensión como 3D R-Tree [15], 2+3 R-Tree [16] y el I+3 R-Tree [17]; métodos en los que el tiempo se incorpora como información dentro de la estructura de los nodos, como el RT-Tree [18]; y métodos que usan múltiples versiones de

la misma estructura para distintos instantes de tiempo, con reutilización de las partes invariantes, como el HR-Tree [16], [19] y el MR-Tree [18].

Para este caso de estudio se propone la utilización del índice I+3 R-Tree que permite manipular puntos en movimiento, administra el tiempo como otra dimensión y responde a las consultas timeslice, intervalo, evento y trayectoria. Es una variante del 2+3 R-Tree en la que se reemplaza el 2D R-Tree de la estructura original, por un Índice.

La estructura del I+3 R-Tree (Figura 1) está formada por:

- Un R-Tree de tres dimensiones (3D R-Tree) que se utiliza para almacenar los cubos cerrados que representan las posiciones anteriores de los objetos y su tiempo de estadía en dicha posición. Cuando un objeto cambia su posición actual, se actualiza su posición en el índice, y se inserta en el 3D R-Tree un cubo que representa su posición anterior.
- Una estructura denominada I (índice) en la que se almacenan los cubos abiertos, es decir, aquellos para los cuales su instante final en una posición aún no está definida. También se guardan las referencias necesarias a los cubos anteriores que describen la trayectoria del objeto.

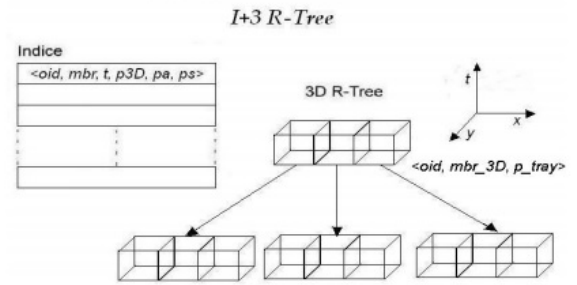


Fig. 1: I+3 R-Tree

Fig. 1. Estructura del I+3 R-Tree

III. APLICACIÓN PARA LA GESTIÓN DE MÓVILES EN LA ATENCIÓN DE EVENTOS DE EMERGENCIA

Se presenta el desarrollo de una aplicación para la atención de eventos de emergencia y la administración de los móviles correspondientes del Municipio de la localidad de Córdoba, Argentina. Este trabajo se desarrolló en el marco de una tesis de Maestría.

Los requerimientos originales para la aplicación fueron:

- Se requiere registrar las llamadas telefónicas realizadas a la línea 911.
- Cada llamada puede generar más de un evento si requiere la intervención de distintas áreas.
- Se requiere definir dos áreas: Emergencia Médica y Emergencia Urbana, con sus correspondientes despachos, que serán las oficinas encargadas de la asignación de los móviles y del seguimiento del evento hasta su finalización.
- Al recibirse una llamada se generan el o los eventos asociados, o también puede asociarse a un evento ya

existente, ya que pueden recibirse varias llamadas por el mismo hecho.

- Durante un evento y en la finalización del mismo se requiere conocer las trayectorias que han realizado los móviles afectados a dicho evento.
- Una vez finalizado un evento se requiere que los móviles afectados al mismo recuperen el estado de "disponible".

En este proyecto se utilizaron dos modelos de bases de datos para almacenar distintos tipos de datos: una base de datos relacional (Figura 2), en la que se registra información estructurada, y una base de datos espacio-temporal (BDET) (con la implementación del índice I+3 R-Tree para realizar consultas de manera eficiente), en la que se registra la información relacionada con la ubicación y tiempos asociados al dinamismo de los móviles.

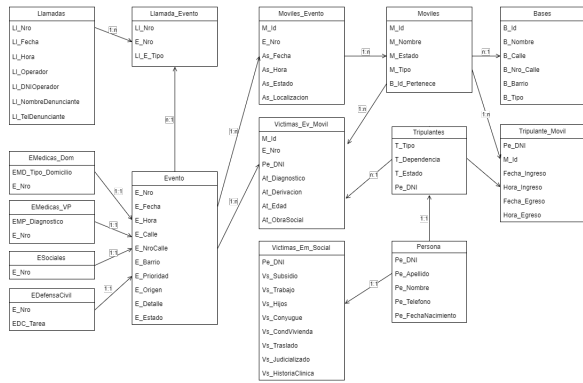


Fig. 2. Base de Datos Relacional

Con la utilización de la Base de Datos Espacio-Temporal en esta aplicación se implementaron los siguientes tipos de consultas:

- Consulta de Trayectoria: permite conocer la trayectoria que realizó un móvil en la atención de un evento, desde que sale de su base, hasta el lugar del evento y luego al destino correspondiente (hospital para los casos en que se registran personas heridas).
- Consulta Instantánea: dado un evento, permite determinar su área de influencia y conocer todos los móviles disponibles que pueden ser asignados para atender dicho evento.
- Consulta por Intervalo: permite obtener por ejemplo todos los móviles asignados a la atención de un evento, conociendo su posición en un intervalo de tiempo.
- Consulta de Historial de Eventos: permite conocer los eventos que ocurrieron en una región en un momento determinado.
- Consulta de los k vecinos más cercanos: dado un móvil y conocido el lugar de un evento, se puede obtener el listado de los móviles más cercanos al mismo para un instante o intervalo de tiempo, para aquellos casos en los que se requiere presencia de más de un móvil para atender el evento.

El desarrollo de la aplicación fue hecho en Java, se uti-

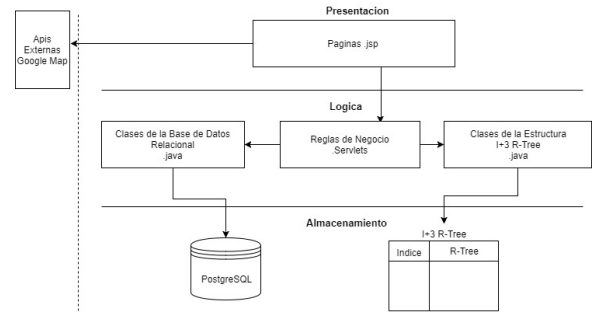


Fig. 3. Arquitectura de la aplicación

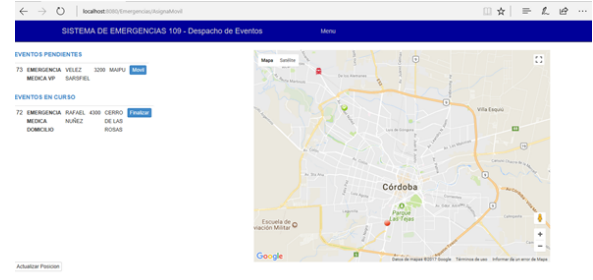


Fig. 4. Vista del despacho con los eventos pendientes

lizó como Sistema Gestor de Bases de Datos Relacional PostgreSQL y se implementó el método de acceso espacio-temporal I+3 R-Tree a través de sus librerías en Java (Figura 3).

El procedimiento al que se da soporte con esta aplicación consiste básicamente en recibir una llamada a través de los operadores telefónicos, quienes la registran y derivan a un despacho.

Luego de registrada la llamada, ésta se puede derivar al despacho si requiere la intervención de algún móvil o finalizarla. En el despacho se cuenta con una vista de los eventos pendientes y en curso con su correspondiente localización en el mapa (Figura 4).

Para la asignación de un móvil a un evento pendiente se obtiene de la tabla de Móviles de la base de datos relacional aquellos que están en estado "disponible" y se los representa gráficamente sobre un mapa, formando con ellos un diagrama de Voronoi [2] (Figura 5).

De esta forma los despachantes pueden ubicar rápidamente cual es el móvil más cercano al evento que están atendiendo. Una vez que se asigna un móvil a un evento pendiente, dicho móvil se incorpora a la vista del despacho y se actualizan los eventos pendientes y en curso (Figura 6). Una vez finalizada la atención se cierra el evento y se libera el móvil, pasando a estar disponible para cubrir un nuevo evento. Periódicamente se registra la posición de los móviles (Figura 7), lo que permite posteriormente reconstruir su trayectoria en la atención de un determinado evento. En las Figuras 8 y 9 se muestran el diagrama del caso de uso y la vista de pantalla de la aplicación correspondientes al proceso de consulta de trayectoria.

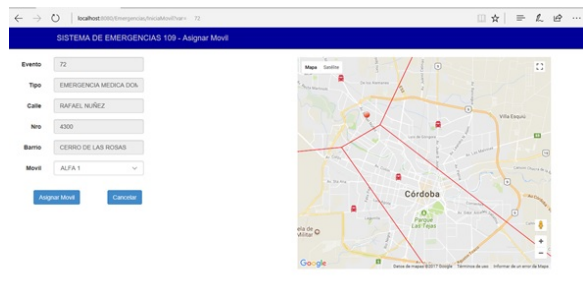


Fig. 5. Vista del despacho con los móviles disponibles

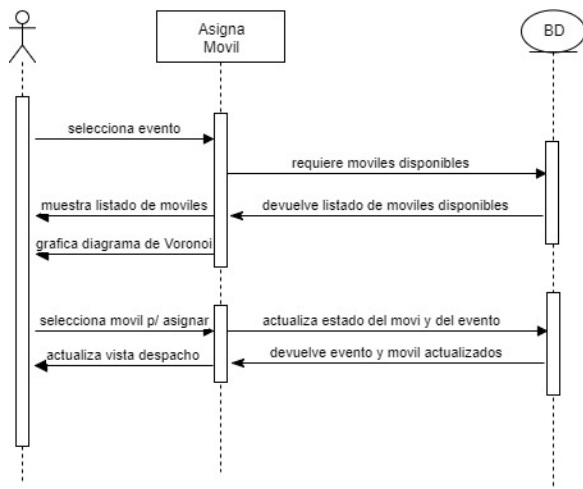


Fig. 6. Proceso de asignación de un móvil

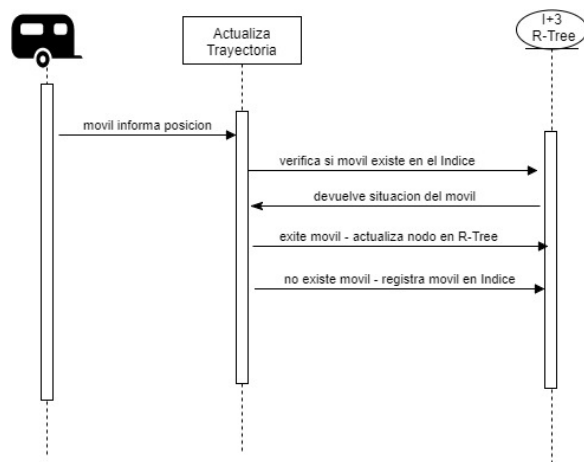


Fig. 7. Reporte de la trayectoria de un móvil

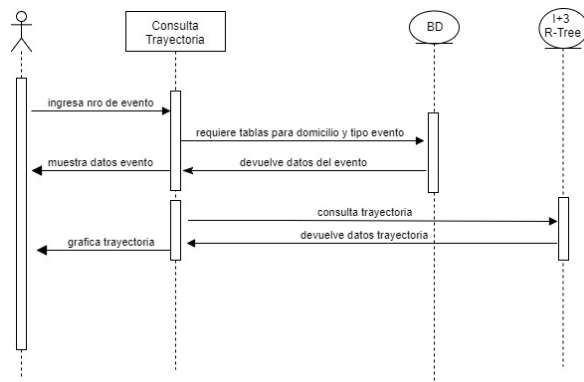


Fig. 8. Proceso que Consulta la Trayectoria de un móvil

El evento registrado puede ser de uno de estos tipos:

- 1) *Emergencia Médica Domiciliaria*: se requiere una ambulancia para la atención de un paciente en su domicilio particular o en alguna institución pública dependiente de la Municipalidad (ej: escuelas, oficinas públicas). En este caso se registra el tipo de domicilio donde se requiere la atención.
- 2) *Emergencia Médica en la Vía Pública*: se solicita una ambulancia para la atención de personas en la vía pública.
- 3) *Emergencia de Defensa Civil*: eventos que requieren la intervención de una cuadrilla para algún trabajo de emergencia pero que no involucra a personas. Por ejemplo: árbol por caer, hundimiento de un pozo, panal de abejas, cable cortado, derrame de líquidos contaminantes, etc.

El I+3 R-Tree está conformado por dos estructuras un 3D R-Tree y un Índice. Las tuplas en el 3D RTree están compuestas por $\langle oid, mbr_{3D}, p_{tray} \rangle$

- *oid*: código identificador del objeto.
- *mbr_{3D}*: región tridimensional cuya altura representa el intervalo temporal durante el cual el objeto se mantuvo en la posición espacial definida por su base.
- *p_{tray}*: puntero al cubo anterior correspondiente al mismo *oid*, utilizado para mantener un historial de trayectoria.

Las tuplas en el índice son de la forma $\langle oid, mbr, t, p3D, pa, ps \rangle$

pa, ps>

- *mbr*: región aproximada que ocupa actualmente el objeto.
- *t*: tiempo de llegada del objeto a su ubicación actual.
- *p3D*: puntero al cubo anterior correspondiente al mismo *oid*, utilizado para mantener un historial de trayectoria.
- *pa*: puntero al objeto insertado en el instante de tiempo inmediatamente anterior.
- *ps*: puntero al objeto insertado en el instante de tiempo siguiente.

Inicialmente en el I+3 R-Tree se registran los móviles con su posición inicial (el 3D R-Tree vacío) (Tabla I).

Al recibir información de una nueva posición del móvil *om1* se actualizan el Índice y el 3D R-Tree (Tablas II y III).

Para la visualización y georreferenciación de los móviles sobre el mapa de la ciudad de Córdoba se utilizaron las APIs de Google Maps [20]. De esta manera se pueden localizar en el mapa de la ciudad los puntos donde se encuentran los eventos en curso y pendientes de atención. En el caso del despacho, en la pantalla correspondiente a la asignación de un móvil para un evento, se implementó un Diagrama de Voronoi, como se puede visualizar en la Figura 5.

Esta estructura geométrica subdivide un área en regiones tomando como base una serie de puntos. Dichas regiones

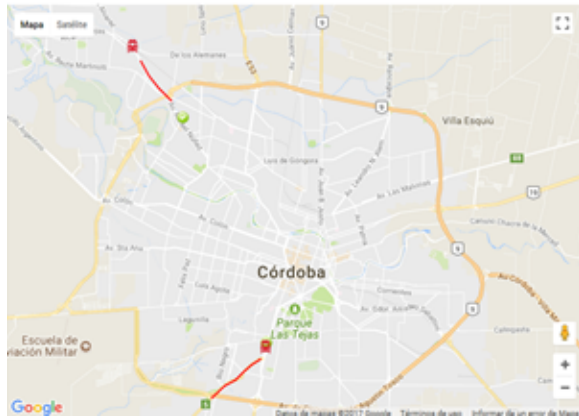


Fig. 9. Vista de una Consulta de Trayectoria

quedan formadas por los puntos más próximos a cada uno de los puntos de base [2]. Para este caso el área a dividir es el mapa de la ciudad de Córdoba y los puntos bases son los móviles disponibles. De esta forma se pueden identificar las distintas áreas de cobertura de cada móvil.

Como en el prototipo de la aplicación no se dispone de una conexión real con los móviles para que transmitan cada determinado período su posición, se realizó una simulación, registrando en el I+3 R-Tree distintos tiempos y posiciones (latitud, longitud) para algunos móviles. Luego de finalizada la atención, en base a los datos registrados en la estructura espacio-temporal se puede resolver la consulta de trayectoria de un móvil, que es uno de los principales requerimientos de información para este sistema.

Tabla I
ÍNDICE

Oid	Mbr	t	p3D	Pa	Ps
om1	om1_pos0	om1_t0	null	null	punt a om2
om2	om2_pos0	om2_t0	null	punt a om1	punt a om3
om3

Tabla II
ÍNDICE

Oid	Mbr	t	p3D	Pa	Ps
om1	om1_pos1	om1_t1	punt al primer cubo del R-Tree	null	punt a om2
om2	om2_pos0	om2_t0	null	punt a om1	punt a om3
om3

Tabla III
3D R-TREE

Oid	mbr_3D	p_tray
om1	posicion0 - Desde tiempo0 hasta tiempo1	null

IV. CONCLUSIONES

Con el desarrollo de esta aplicación se pudo probar que la combinación del índice espacio-temporal I+3 R-Tree con una base de datos relacional provee una solución adecuada para la gestión y asignación de objetos móviles implicados en la atención de eventos de emergencias médicas y urbanas. Teniendo en cuenta las características de los móviles que se pretende gestionar con esta aplicación se consideró el índice espacio-temporal I+3 R-Tree como el más conveniente de implementar ya que permite administrar objetos con las siguientes características:

- Los objetos se mueven a gran velocidad.
- El tamaño y la forma de los objetos no son importantes, el interés se centra en su posición en el tiempo.
- El espacio no cambia y la cantidad de objetos no tiene mucha variación.
- Los objetos se mueven en áreas previamente establecidas.
- Se necesita consultar la trayectoria realizada por los objetos.

Una de los requerimientos más importantes de la aplicación es la posibilidad de resolver la consulta de trayectoria, que pudo ser resuelta eficientemente a partir de la implementación del índice espacio-temporal seleccionado. Los resultados de las primeras pruebas son alentadores, dado que con la aplicación se resuelven de manera satisfactoria los requerimientos planteados: permite el registro de las llamadas efectuadas al 911 del Municipio de Córdoba; a partir de una llamada se puede generar más de un evento y asignar los móviles disponibles más cercanos al lugar del evento registrado; una vez finalizado el evento se puede reconstruir la trayectoria realizada por cada móvil. Como trabajo futuro se prevé la implementación de nuevas consultas, en particular, del Join Espacio-Temporal.

REFERENCIAS

- [1] R. H. Güting and M. Schneider, *Moving objects databases*. Morgan Kaufmann, 2005.
- [2] M. d. Berg, O. Cheong, M. v. Kreveld, and M. Overmars, *Computational Geometry: Algorithms and Applications*, 3rd ed. Santa Clara, CA, USA: Springer-Verlag TELOS, 2008.
- [3] S. Shekhar and S. Chawla, *Spatial databases : a tour*. Prentice Hall, 2003. [Online]. Accesible en: <https://goo.gl/1WPSHZ>
- [4] Y. Theodoridis, T. Sellis, A. Papadopoulos, and Y. Manolopoulos, "Specifications for efficient indexing in spatiotemporal databases," in *Proceedings. Tenth International Conference on Scientific and Statistical Database Management (Cat. No.98TB100243)*. IEEE Comput. Soc, 1998, pp. 123–132. [Online]. Accesible en: <https://goo.gl/DK5Ljm>
- [5] V. Gaede and O. Güting, "Multidimensional access methods," *ACM Computing Surveys*, vol. 30, no. 2, pp. 170–231, jun 1998. [Online]. Accesible en: <https://goo.gl/LGtX1>
- [6] G. A. G. Retamal, "Metodos de acceso y procesamiento de consultas espacio-temporales," Ph.D. dissertation, 2007. [Online]. Accesible en: <https://goo.gl/gKmyCG>
- [7] B. Stantic, J. Thornton, and A. Sattar, "A novel approach to model NOW in temporal databases," *10th International Symposium on Temporal Representation and Reasoning, 2003 and Fourth International Conference on Temporal Logic. Proceedings.*, 2003.
- [8] A. C. De Battista, "Búsquedas por similitud sobre objetos dinámicos," nov 2008. [Online]. Accesible en: <https://goo.gl/FbhcV>

- [9] C. Dyreson, B. Pernici, J. F. Roddick, N. L. Sarda, M. R. Scalas, A. Segev, R. T. Snodgrass, M. D. Soo, A. Tansel, P. Tiberio, G. Wiederhold, F. Grandi, W. K?fer, N. Kline, N. Lorentzos, Y. Mitsopoulos, A. Montanari, D. Nonen, E. Peressi, A. Segev, R. T. Snodgrass, M. D. Soo, A. Tansel, P. Tiberio, and G. Wiederhold, "A consensus glossary of temporal database concepts," *ACM SIGMOD Record*, vol. 23, no. 1, pp. 52–64, mar 1994. [Online]. Accesible en: <https://goo.gl/B6XfLh>
- [10] M. F. Mokbel, T. M. Ghanem, and W. G. Aref, "Spatio-temporal access methods," *IEEE Data Engineering Bulletin*, vol. 2, no. 26, pp. 40–49, 2003. [Online]. Accesible en: <http://www-users.cs.umn.edu/~mokbel/papers/STIndex.pdf>
- [11] M. Schneider, "SPATIAL AND SPATIO-TEMPORAL DATA MODELS AND LANGUAGES." [Online]. Accesible en: <https://goo.gl/F8cKDT>
- [12] H.-K. Ahn, N. M. Ho, M. Wong, H. K. Ahn, N. Mamoulis, and H. M. Wong, "A Survey on Multidimensional Access Methods A Survey on Multidimensional Access Methods *," 2001. [Online]. Accesible en: <https://goo.gl/LmLdTW>
- [13] P. K. Agarwal, L. Arge, and J. Erickson, "Indexing moving points (extended abstract)," in *Proceedings of the Nineteenth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, ser. PODS '00. New York, NY, USA: ACM, 2000, pp. 175–186. [Online]. Accesible en: <https://goo.gl/MfUUXP>
- [14] A. Guttman, "R-trees: A dynamic index structure for spatial searching," *SIGMOD Rec.*, vol. 14, no. 2, pp. 47–57, Jun. 1984. [Online]. Accesible en: <https://goo.gl/CuAEB8>
- [15] Y. Theodoridis, M. Vazirgiannis, and T. Sellis, "Spatio-Temporal indexing for large multimedia applications," *Multimedia Computing and Systems, International Conference on*, vol. 0, pp. 0441+, 1996. [Online]. Accesible en: <https://goo.gl/MXCfgL>
- [16] M. A. Nascimento, J. R. O. Silva, and Y. Theodoridis, "Evaluation of Access Structures for Discretely Moving Points," *Proceedings of the International Workshop on Spatio-Temporal Database Management*, pp. 171–188, 1999. [Online]. Accesible en: <https://goo.gl/MHBy23>
- [17] E. O. Gagliardi, F. D. Carrasco, and J. C. García Sosa, "I+3 R-Tree: un método de acceso espacio-temporal," 2009. [Online]. Accesible en: <https://goo.gl/mMPShC>
- [18] X. Xu, J. Han, and W. Lu, "RT-tree: An improved R-tree index structure for spatio-temporal database," *4th International Symposium on Spatial Data Handling*, pp. 1040—1049, 1990.
- [19] M. A. Nascimento, J. R. O. Silva, Y. Theodoridis, C. S. Jensen, M. H. Böhlen, R. Busatto, C. E. Dyreson, H. Gregersen, D. Pfoser, S. Saltenis, J. Skyt, G. Slivinskas, K. Torp, R. T. Snodgrass, S. Ram, M. D. Soo, A. Steiner, and S. Timeconsult, "Access Structures for Moving Points," 1998. [Online]. Accesible en: <https://goo.gl/isUr2A>
- [20] Api google maps. <http://goo.gl/QH7y4K>.